

**REMARKS**

**A. STATUS OF CLAIMS**

Claims 1-22 are pending and stand rejected.

**B. PREMATURE FINAL**

The Examiner is respectfully requested to withdraw the finality of the last Office action on the grounds that it was premature (see MPEP 706.07(a), 706.07(c) and 706.07(d)). MPEP 707.07(a) defines a premature final as one "where the examiner introduces a new ground of rejection not necessitated by amendment of the application by applicant". The rejections of claims 1 and 8, discussed below, under 35 U.S.C. §112, second paragraph, was not necessitated by amendment and therefore, the finality of the last Office Action is deemed premature.

**C. Claims 1 and 8 were rejected under 35 U.S.C. §112, second paragraph.** The Applicant respectfully traverses this rejection for the following reason(s).

Claim 1 calls for, in part, *input means for reading out the product key information from the memory means and inputting the read-out product key information in an information input window for product certification of the operating system program when a product key of an operating system program being reinstalled is matched with the read-out product key information.*

The Examiner, for some reason, now finds the forgoing feature to be "unclear" after three previous office actions. This is not understood.

As for the foregoing feature, the Examiner indicates that the *input means for reading out the product key information* is contradictory in that it implies an output operation conducted by an input means. The Examiner appears to be examining the feature in a piecemeal manner. The feature should be examined as it is wholly written. Clearly, as written calls for *input means for . . . inputting the read-out product key information in an information input window . . .*, wherein the *inputting* feature clearly supports the tag word *input* (input means).

The Examiner also now finds that it is not clear whether the *input means*, which has read the product key info, then inputs the info in the *information input window*, or if the user inputs the info into the *information input window*. Since there is no 'comma' in the sentence of the foregoing feature, then it is clear that the *input means* has the function of *inputting the read-out product key information in an information input window*.

Additionally, it is required that the claim is to be read in light of the specification, which clearly supports the claim as written.

Further, the Examiner also holds that claim 1 seems to be missing a step or structure causing ambiguity and overall vagueness.

There are no missing steps or structure necessary to the invention set forth in claim 1.

Accordingly, the rejection of claim 1 is deemed to be in error and should be withdrawn.

Claim 8 calls for, in part, *checking whether the read-out product key information is matched*

*with product key information of an operating system program that will be reinstalled.*

The Examiner, for some reason, now finds the forgoing feature to be "unclear" after three previous office actions. This is not understood.

As for the foregoing feature, the Examiner indicates that the source of the product key info of an operating system that will be reinstalled is not mentioned in the limitation of the claim.

First, note that the preamble calls for *In a computer system having an operating system program containing product key information and comprising a central processing unit, a main memory, an auxiliary memory having the product key information . . .* Second, the claim further requires *reading out the product key information from the auxiliary memory.*

Accordingly, the source of the *read-out product key information* is *an auxiliary memory*. Therefore, it is very clear that the source of the *read-out product key information* is *an auxiliary memory*.

Additionally, the Examiner indicates that there seems to be a missing comparing step between the "reading out" step and the "checking" step. The Examiner's indication is not understood, and might be better understood what the Examiner means if the Examiner wrote out the step completely to indicate what was being compared to what.

In the meantime, note that the step of *checking whether the read-out product key information is matched with product key information of an operating system program that will be reinstalled* is a comparing step wherein the *read-out product key information* is compared to *product key information of an operating system program that will be reinstalled* to see if they match. It is not understood why there should be a further comparing step claimed, as implied by the Examiner, since

the *checking* step is a comparing step. The Applicant can be his own lexicographer.

Further, the Examiner refers to the claims preamble wherein it is noted that there is mention of manually inputting a product key by a user, but then suggests that the "missing step has not been given patentable weight because the recitation occurs in the preamble." It is not understood why 'patentable weight' is being considered under §112, as it is supposed to be a basis of consideration under §102 or §103.

Considering the Examiner's rejection, however, note that it is not understood how a step can be missing if it is in the preamble. Additionally, it is not understood if the "missing step" is something in the preamble or the "comparing step" mentioned previously.

If it is the "comparing step", then, as discussed above, there is already a comparing step which calls for *checking whether the read-out product key information is matched with product key information of an operating system program that will be reinstalled*.

As for the feature in the preamble, it is stipulated that a computer system has an auxiliary memory having the product key information, manually input by a user when the operating system program was first installed, stored therein. Accordingly such stipulation **does** require consideration because it provides patentable weight to the claimed feature of *reading out the product key information from the auxiliary memory*.

It has long been an accepted practice in the PTO to have the preamble give meaning to the claim and properly define the invention, *Gerber Garment Technology, Inc. v. Lectra Systems, Inc.*, 916 F.2d 683, 16 USPQ 2d 1436, 1441 (Fed. Cir. 1990). In *Gerber Garment Technology, Inc. v.*

*Lectra Systems, Inc.*, claim 15 was reviewed wherein claim 15 used the terms "a cutting blade" and "an object" in the preamble. The body of claim 15 referred back to the "cutting blade" and the "object" by preceding these features with the terms "said" or "the". There was no rejection offered by either the Examiner, the U.S. Patent and Trademark Board of Appeals and Interferences, nor the Court of Appeals for the Federal Circuit, under 35 U.S.C. §112, second paragraph which would have invalidated the claim. Accordingly, the absence of such a rejection of claim 15 under §112, second paragraph is proof of that referring back to terms in the preamble by preceding the terms with "said" or "the" when there is proper antecedent basis for the term in the preamble, is an acceptable claim drafting practice.

In *Derman v. PC Guardian*, 37 USPQ2d 1733 (CAFC 1995) it was determined that a term or phrase in the preamble, when referenced in the body of the claim by "said" or "the", provides necessary definition for structural limitations.

Also, see *Loctite Corp. v. Ultraseal Ltd.*, 781 F.2d 861, 228 USPQ 90 (Fed. Cir. 1985) held that a term in the preamble of a claim that "breathes life and meaning" into the subject matter of that claim is a necessary limitation on the scope of the claim; and *DeGeorge v. Bernier*, 768 F.2d 1318, 226 USPQ 758 (Fed. Cir. 1985) held that generally, the preamble does not limit a claim unless preamble limitations are necessary to give meaning to the claim and properly define the invention.

The preamble gives meaning to the claim because it identifies the auxiliary memory and how product key information comes to be stored therein, *i.e.*, by being manually input by a user when the operating system program was first installed in the computer system.

Further, the rejection includes a statement "The noted recitation is found in the preambles of claims 1, 7, 13 and 14." The Applicant is not sure what the Examiner means by the phrase "noted recitation", especially considering that none of claims 7, 13 and 14 contain a preamble.

The Applicant respectfully requests that the Examiner be more specific with regard to what is being referred to.

Further yet, the Examiner also holds that claim 8 seems to be missing a step or structure causing ambiguity and overall vagueness, similar to claim 1.

As noted with respect to claim 1, there is no missing step or structure.

Accordingly, the rejection of claim 8 is deemed to be in error and should be withdrawn.

**D. Claims 1, 2, 4, 5, 8-12, 21 and 22 were rejected under 35 U.S.C. §103(a), as rendered obvious and unpatentable, over Hoggarth et al. (US 6,535,976) in view of "Aptivia" ("Installing Windows 98 on an Aptiva 2168 system") in further view of Krosner et al. (US 5,905,494). The Applicant respectfully traverses this rejection for the following reason(s).**

#### **Claim 1**

Claim 1 calls for a *memory means storing BIOS setup information* separate from a BIOS ROM, *said memory means storing the product key information of the operating system.*

As noted in the Amendment filed 8/31/04 none of the references disclose or teach of such

*a memory means.*

In paragraph 13 starting on page 22 of the final rejection the Examiner holds "a system that inputs product key information must inherently store this information. It is well known in the art that the Windows™ operating system stores product key information in a registry file. The Information is inherently stored somewhere."

The Applicant disagrees. If the system product key for the Windows™ operating system were stored in the registry file, then anyone could go in to the registry file and look the information up. Microsoft is notoriously well known for trying to keep certain information secret, such as the product key for each operating system, in order to prevent piracy.

If it is well known, then the Examiner should provide a reference that teaches storing the Windows operating system's product key information on a personal computer. See MPEP §2144.03

As described in the Applicant's specification, BIOS ROM 106 stores a BIOS program that controls the booting operation of the computer system using HDD 140 or CD-ROM 160 in accordance with the information stored in CMOS RAM 108. The CMOS RAM 108 always retains the BIOS setup information.

There is no teaching of such a memory means as claimed, and especially a *memory means storing BIOS setup information used by a BIOS program stored in said BIOS ROM, said memory means storing the product key information of the operating system.*

The Examiner's suggestion of inherency is in error. *In Re Oelrich*, 666 f.2d 578# 581-82, 212 USPQ 323,326 (CCPA 1981) "The mere fact that a certain thing *may* result from a given set of

circumstances is not sufficient [to establish inherency.]" *In re Rijckaert*, 28 USPQ 1953, 1957 (CAFC 1993) "a retrospective view of inherency is not a substitute for some teaching or suggestion supporting an obviousness rejection", citing *In re Newell*, 891 F.2d 899,901, 13 USPQ2d 1248, 1250 (Fed. Cir. 1989).

Claim 1 also calls for *input means for reading out the product key information from the memory means and inputting the read-out product key information in an information input window for product certification of the operating system program when a product key of an operating system program being reinstalled is matched with the read-out product key information.*

Aptiva discusses initial installation of a new operating system (Windows 98) over an old operating system (Windows 95). Since all Microsoft operating system software has its own unique product key, it would make no sense to read product key information of the Windows 95 operating system when installing a new operating system that has its own different product key.

It is **well known in the art**, when installing Microsoft operating system software to display a window for **the user to enter** the product key information. It is also well known in the art that this product key will never be displayed again.

It appears that the Examiner finds the foregoing feature of claim 1 (see also claim 8) to be deficient under 35 U.S.C. §112, second paragraph, in order to maintain the rejection based on holding that the user inputs the product key information instead of the claimed *input means*.

Since there is no 'comma' in the sentence of the foregoing feature, then it is clear that the *input means* has the function of *inputting the read-out product key information in an information*

*input window*, not a user.

Further, claim 1, and similarly claim 8, calls for inputting the read-out product key information in an information input window for product certification of the operating system program *when a product key of an operating system program being reinstalled is matched with the read-out product key information*. Accordingly, it is necessary to compare the product key stored in the auxiliary memory with the product key information stored on the operating system's installation disk to determine that there is a match.

None of the references teach looking for a match between data prior to displaying the data.

The Examiner fails to address this difference between the claimed invention and the applied prior art. Accordingly, the Examiner is respectfully requested to fully respond to this traversal by explaining how the applied art teaches *input means for reading out the product key information from the memory means and inputting the read-out product key information in an information input window for product certification of the operating system program when a product key of an operating system program being reinstalled is matched with the read-out product key information.*

Accordingly, the rejection of claim 1 is deemed to be in error for the foregoing reasons, as well as the reasons set forth in the traversal in the Amendment filed 8/31/04, and thus should be withdrawn.

## Claim 2

Claim 2 requires that the *storing means* be a *boot device*. Here the Examiner refers to a portion of Hoggarth indicating that Hoggarth has a memory and a boot mechanism.

The Examiner fails, however, to explain how one of ordinary skill in the art would have been motivated to use Hoggarth's memory as a *boot device*, or why one of ordinary skill in the art would have used Hoggarth's memory to store *BIOS setup information* and *the product key information of the operating system*.

Accordingly, the rejection of claim 2 is deemed to be in error and should be withdrawn.

#### Claim 4

Claim 4 stipulates that *the writing means is a program installed in the storing means*. Here the Examiner refers to a portion of Hoggarth, *i.e.*, col. 9, lines 25-38, which state:

In one example, the maintenance program comprises an operating system upgrade; which may be either a complete operating system or additional upgrade code to add to the existing software. Thus, the server downloads a revised operating system and install program which replaces the operating system currently stored on the hardfile. Alternatively, the maintenance software can be an upgraded version of the client system BIOS which is re-written over the original BIOS. The upgraded BIOS is written directly over the existing BIOS in flash ROM, while preserving any client-specific information such as serial number and model number. The server advantageously invokes an immediate reboot of the client after the upgraded BIOS is loaded in order to make use of the new level of system BIOS.

Contrary to the Examiner's rejection, the cited section of Hoggarth fails to make mention of **any product key information**, much less the required *operating system program containing product key information*, and thus fails to teach *writing means for writing the product key information in the*

*memory means*, as required by claim 1.

Accordingly, the rejection of claim 4 is deemed to be in error and should be withdrawn.

#### **Claim 5**

Claim 5 stipulates that *the input means is a program*. Here the Examiner refers to a portion of Hoggarth, *i.e.*, col. 9, lines 25-38 (see the rejection of claim 4, above).

Contrary to the Examiner's rejection, the cited section of Hoggarth fails to make mention of **any product key information**, much less the required *operating system program containing product key information*, and thus fails to teach *input means for reading out the product key information from the memory means and inputting the read-out product key information in an information input window*, as required by claim 1.

Accordingly, the rejection of claim 5 is deemed to be in error and should be withdrawn.

#### **Claim 8**

Claim 8 calls for a computer system having an operating system program containing product key information and comprising a central processing unit, a main memory, an auxiliary memory having the product key information, manually input by a user when the operating system program was first installed, stored therein. And is directed towards a method of automatically re-inputting the product key information when reinstalling the operating system program.

In particular, the method of claim 8 requires *reading out the product key information from the auxiliary memory*. The Examiner notes that the foregoing feature is not found in Hoggarth nor

Aptiva, and applies Krosner in this regard.

The Examiner suggests that Krosner's teaching of the display of several possible entries for selection by a user when prompted to input appropriate information into an input field (col. 1, lines 33-53) be referred to as "a read-out field." However, the claim calls for reading out from an auxiliary memory, not from a "read-out field."

Accordingly, the rejection of claim 8 is deemed to be in error and should be withdrawn.

Additionally, claim 8 requires the product key information read-out from the auxiliary memory be checked with product key information of an operating system program that will be reinstalled to determine if they match, and *if matched, automatically inputting the product key information in a product key information input window displayed on a screen corresponding to an installation procedure for reinstalling the operating system program.*

The Examiner refers to user input by suggesting that the user can "retrieve information from the system without referring to any other resources . . . ." However, the Examiner has not indicated how such retrieval is automatic, nor identified where information was prior to be retrieved by the user.

Also, it is required by claim 8 that product key information read-out from the auxiliary memory be checked with product key information of an operating system program that will be reinstalled to determine if they match. The Examiner has not identified where such a feature (*checking whether the read-out product key information is matched with product key information of an operating system program that will be reinstalled*) is taught by the applied art. And then *if*

*matched*, the product key information will automatically be input in a product key information input window displayed on a screen.

Accordingly, the rejection of claim 8 is deemed to be in error and should be withdrawn.

#### **Claim 11**

Claim 11 calls for *writing said product key information into a predetermined storage area of a CMOS RAM*.

The Examiner fails to identify which, if any, of the applied references disclose a *CMOS RAM*, much less teach writing *product key information into a predetermined storage area* thereof. Note, *Ex parte Levy*, 17 USPQ2d 1461, 1462 (1990) states:

"it is incumbent upon the examiner to identify wherein each and every facet of the claimed invention is disclosed in the applied reference."

Claim 11 also calls for *reading out said product key information from said CMOS RAM when a recovery program is executed*. The Examiner fails to identify where this feature is taught by the applied art.

Additionally, claim 8 calls for *comparing said product key information read out from said CMOS RAM with product key information stored in said recovery storage device*. The Examiner fails to identify where this feature is taught by the applied art.

Further, claim 11 calls for *automatically inputting the product key information read out from said CMOS RAM into said product key input window of the product key input screen displayed on said display device.*

It appears that the Examiner is applying Krosner in this regard by suggesting that the user can "retrieve information from the system without referring to any other resources . . . " However, the Examiner has not indicated how such retrieval is automatic, nor identified where information was prior to be retrieved by the user.

In paragraph 13 starting on page 23 of the final rejection the Examiner holds "a system that inputs product key information must inherently store this information. It is well known in the art that the Windows™ operating system stores product key information in a registry file. The Information is inherently stored somewhere."

The Applicant disagrees. If the system product key for the Windows™ operating system were stored in the registry file, then anyone could go in to the registry file and look the information up. Microsoft is notoriously well known for trying to keep certain information secret, such as the product key for each operating system, in order to prevent piracy.

If it is well known, then the Examiner should provide a reference that teaches storing the Windows operating system's product key information on a personal computer. See MPEP §2144.03

The Examiner's suggestion of inherency is in error. *In Re Oelrich*, 666 f.2d 578# 581-82, 212 USPQ 323,326 (CCPA 1981) "The mere fact that a certain thing *may* result from a given set of circumstances is not sufficient [to establish inherency.]" *In re Rijckaert*, 28 USPQ 1953, 1957 (CAFC 1993) "a retrospective view of inherency is not a substitute for some teaching or suggestion

supporting an obviousness rejection", citing *In re Newell*, 891 F.2d 899,901, 13 USPQ2d 1248, 1250 (Fed. Cir. 1989)..

Accordingly, the rejection of claim 11 is deemed to be in error and should be withdrawn.

#### **Claim 12**

Claim 12 calls for a step of *storing said product key information manually input into said product key input window onto said hard disk.*

The Examiner fails to identify which, if any, of the applied references discloses storing *product key information onto a hard disk.* See *Ex parte Levy*, supra.

**E. Claims 3, 13 and 15-19 were rejected under 35 U.S.C. §103(a), as rendered obvious and unpatentable, over Hoggarth, Aptiva and Krosner as applied to claims 1 and, an in further view of Micali (US 5793868).** The Applicant respectfully traverses this rejections for the following reason(s).

As noted above, several features of claims 1 and 11, from which claims 3, 13 and 15-19 depend, are lacking in the combined teachings of the applied art. Micali fails to provide any teaching regarding those features noted as lacking in the combined teachings of the applied art.

Therefore, since claims 3, 13 and 15-19 incorporate the features of claims 1 and 11, by definition of a dependent claim, then the rejections of claims 3, 13 and 15-19 is deemed to be in error

for the same reasons discussed above with respect to claims 1 and 11. Accordingly, the rejections should be withdrawn.

### **Claim 3**

Claim 3 calls for the memory means to further store *information indicating the type of operating system program that was installed and indicating a compress conversion process of the product key information.*

The Examiner notes that none of Hoggarth, Aptiva and Krosner teach the foregoing feature of claim 3. Here, the Examiner refers us to Micali, col 4, lines 29-46 and col. 8, lines 6-9 which state:

According further to the present invention, an authority generates authenticated information about revoked certificates by generating data identifying the revoked certificates, generating compressed date information indicating a date of revocation for each of a first subgroup of the revoked certificates that contains at least one of the revoked certificates, and by generating the authenticated information by authenticating at least one of: the data together with the compressed date information alone, the data together with compressed date information and other date information, and the date together with the compressed date information and other information. The other date information may include revocation dates of the revoked certificates that are outside of the first subgroup. Generating compressed date information may include specifying a number of days between a revocation date and a reference date, which may be the date of issuance of the certificate or may be the date of authentication. The first subgroup may contain all of the revoked certificates; and

Alternatively still, it is possible to provide compact structures for conveying certificate revocation information that are more efficient than conventional CRLs even while retaining revocation date

information (or other information).

Accordingly, Micali teaches compressing date information pertaining to a revocation date of data that was previously certified but is now revoked.

None of the applied references, Hoggarth, Aptiva and Krosner, either singularly or in combination discuss or teach a need to revoke previously certified data. Accordingly, there is no motivation for one of ordinary skill in the art to look to Micali's teaching pertaining to the field of revoking previously certified data.

Additionally, invention relates generally to secure communications and that a study envisages that each revoked certificate is specified in a CRL by means of about 9 bytes: 20 bits of serial number and 48 bits of revocation date. Thus, in the Federal PKI, each CRL is expected to comprise thousands of certificate serial numbers and their revocation dates; the header, however, has a fixed length, consisting of just 51 bytes. Thus, communication, at two cents per kilobyte, the impact of CRL transmission on the estimated yearly costs of running the Federal PKI is stunning: if each federal employee verifies one hundred digital signatures per day on average, then the total PKI yearly costs are \$10,848 million of which \$10,237 million is due to CRL transmission. If each employee is assumed to verify just five digital signatures a day on average, then the total PKI yearly costs are \$732 million, of which 563 million is due to CRL transmission.

Accordingly, Micali desires to compress the revocation date in order to reduce the cost of communication.

The combined teachings of Hoggarth, Aptiva and Krosner do not call for data communication

that will have any cost. Therefore, there is no need to reduce communication costs in the invention resulting for the combined teachings of Hoggarth, Aptiva and Krosner. Accordingly, there is no motivation for one of ordinary skill in the art to look to Micali for a teaching on how to reduce communication cost.

Additionally, there is no teaching in Micali to motivate one of ordinary skill in the art to compress *product key information*, based on Micali's teaching of a compressed revocation date.

Therefore, the rejection of claim 3 is deemed to be in error and should be withdrawn.

### **Claim 13**

Claim 13 calls for, in part, *encoding said product information using a compression conversion process to produce encoded product key information*. See the traversal of the rejection of claim 3.

### **Claim 15**

Claim 15 stipulates that the compression conversion process of claim 13 comprises the steps of *converting each ASCII character into a six bit code; and generating hexadecimal values for storage in said CMOS RAM by grouping the bits of the six bit codes corresponding to every four ASCII characters into three bytes*.

The Examiner takes Official Notice that it is well known to convert ASCII code to hexadecimal format and mapping ASCII values into x-length string bytes in memory.

The Applicant disagrees. The Applicant also does not believe that such conversion and mapping is a known *compression conversion process*. If it is well known, then the Examiner should provide a reference that teaches such a compression conversion process. See MPEP §2144.03

Although it is well known in the art to convert ASCII to hexadecimal, it is not known to convert each ASCII character into a six bit code, then generate a hexadecimal value by *grouping the bits of the six bit codes corresponding to every four ASCII characters into three bytes*.

Accordingly, the rejection of claim 15 is deemed to be in error and should be withdrawn.

#### **Claim 16**

Claim 16 requires that the conversion of each ASCII character into a six bit code *comprises subtracting the hexadecimal value 30h from the hexadecimal of the ASCII character*.

Again, the Examiner invokes Official Notice, to which the Applicant disagrees. If it is well known, then the Examiner should provide a reference that teaches such a conversion process. See MPEP §2144.03.

Accordingly, the rejection of claim 16 is deemed to be in error and should be withdrawn.

#### **Claim 17**

Claim 17 requires *converting each ASCII character into a six bit code comprises reading preset hexadecimal values for each ASCII character from a code table and changing the read hexadecimal values to binary values*.

Again, the Examiner invokes Official Notice, to which the Applicant disagrees. If it is well

known, then the Examiner should provide a reference that teaches such a conversion process. See MPEP §2144.03.

Accordingly, the rejection of claim 17 is deemed to be in error and should be withdrawn.

#### **Claim 18**

Claim 18 calls for *converting each ASCII character into a five bit code; and generating hexadecimal values for storage in said CMOS RAM by grouping the bits of the five bit codes corresponding to every three ASCII characters into two bytes.*

Again, the Examiner invokes Official Notice, to which the Applicant disagrees. If it is well known, then the Examiner should provide a reference that teaches such a conversion process. See MPEP §2144.03.

Accordingly, the rejection of claim 18 is deemed to be in error and should be withdrawn.

#### **Claim 19**

Claim 19 calls for *converting each ASCII character into a five bit code comprises reading preset hexadecimal values for each ASCII character from a code table and changing the read hexadecimal values to binary values.*

Again, the Examiner invokes Official Notice, to which the Applicant disagrees. If it is well known, then the Examiner should provide a reference that teaches such a conversion process. See MPEP §2144.03.

Accordingly, the rejection of claim 18 is deemed to be in error and should be withdrawn.

**F. Claim 14 was rejected under 35 U.S.C. §103(a), as rendered obvious and unpatentable, over Hoggarth in view of Aptiva, Krosner and Micali as applied to claims 13, and in further view of Miura (US 5,930,505, not US 6,021,408). The Applicant respectfully traverses this rejection for the following reason(s).**

Claim 14 calls for *a step of uninstalling said key input program from said hard disk after said storing step.*

The Examiner has applied Miura as a teaching of deleting a program after execution (col. 9, lines 29-33). The Examiner reasons that it would have been obvious to delete such a key input program "because this combination provides another level of security to the [?] by preventing an unauthorized installation in the case where only one installation is allowed, and frees up memory resources."

Deficiencies in the factual basis cannot be supplied by resorting to speculation or unsupported generalities. *In re Warner*, 379 F.2d 1011, 154 USPQ 173 (CCPA 1967) and *In re Freed*, 425 F.2d 785, 165 USPQ 570 (CCPA 1970).

There has been no showing in the proposed combination of Hoggarth Aptiva, Krosner and Micali of allowing only one installation. Note the [?] above in quoting the Examiner, and note that the Examiner has not mentioned what is being installed, or allowed to be installed, only once.

Additionally, Miura simply discloses that when execution of a user program has been finished, the user program is deleted. Miura gives no reason for the deletions, and in particular fails to teach the reasons, such as security and freeing up memory, suggested by the Examiner.

That a prior art device could be modified to produce the claimed device does not justify an obviousness rejection unless the prior art suggested the modification's desirability. *In re Gordon*, 733 F.2d 900, 221 USPQ 1125 (Fed. Cir. 1984).

Here, the Examiner provides the suggestion, not the prior art.

Accordingly, the rejection of claim 14 is deemed to be in error and should be withdrawn.

Further, Miura fails to provide the necessary information need to support the rejection of claim 11, such information being the teachings noted as lacking in the combination of Colosso, Markus and Sobel above.

Additionally, Miura fails to provide the necessary information need to support the rejection of claim 13, such information being the teachings noted as lacking in the combination of Colosso, Markus, Sobel and Ledain above.

Accordingly, the rejection of dependent claims 14 is deemed to be in error for the same reasons as argued above with respect to the rejection of claims 11 and 13.

**G. Claim 20 was rejected under 35 U.S.C. §103(a), as rendered obvious and unpatentable, over Hoggarth in view of Aptiva and Krosner as applied to claim 11, and in further view of Pearce et al. (US 6,243,468). The Applicant respectfully traverses this rejection for the following reason(s).**

Claim 20 calls for *checking a checksum of said product key information read out from said*

*CMOS RAM before comparing said product key information read out from said CMOS RAM with product key information stored in said recovery storage device.*

Here, the Examiner refers to Pierce's col. 2, lines 44-60 with respect to the *checksum*. Pierce teaches a system that includes a software product that is loaded onto a specific computer having a set of hardware components (e.g., RAM, hard disk drive, floppy disk drive, BIOS, network card, video card, etc.). The software product has an associated product ID consisting of, for example, a manufacturer ID, a registered product code, a serialized number, and a **checksum** value.

Pierce never mentions nor teaches *checking a checksum of said product key information read out from said CMOS RAM*.

Deficiencies in the factual basis cannot be supplied by resorting to speculation or unsupported generalities. *In re Warner*, 379 F.2d 1011, 154 USPQ 173 (CCPA 1967) and *In re Freed*, 425 F.2d 785, 165 USPQ 570 (CCPA 1970).

That a prior art device could be modified to produce the claimed device does not justify an obviousness rejection unless the prior art suggested the modification's desirability. *In re Gordon*, 733 F.2d 900, 221 USPQ 1125 (Fed. Cir. 1984).

Here, the Examiner provides the suggestion, not the prior art.

Accordingly, the rejection of claim 20 is deemed to be in error and should be withdrawn.

Additionally, claim 20 requires that the *checking a checksum of a product key information read out from said CMOS RAM*, not downloaded from a the Internet. Accordingly, the checksum and product key information have already been matched, or else the product key information would

not have been stored in the CMOS RAM.

Further, claim 20 requires that the checking a checksum of a product key information read out from said CMOS RAM before comparing said product key information read out from said CMOS RAM with product key information stored in said recovery storage device.

The applied prior art fails to teach comparing said product key information read out from said CMOS RAM with product key information stored in said recovery storage device, and clearly fails to teach a need to check the checksum before making such a comparison.

Accordingly, the rejection of claim 20 is deemed to be in error and should be withdrawn.

**H. Claim 6 was rejected under 35 U.S.C. §103(a), as rendered obvious and unpatentable, over Hoggarth in view of Aptiva.** The Applicant respectfully traverses this rejection for the following reason(s).

Claim 6 calls for a step of *writing the manually input product key information into the auxiliary memory.*

The Examiner notes that Hoggarth fails to disclose storing the product key in an auxiliary memory, and notes that Aptiva discloses a product key for an operating system, but does not provide any *prima facie* showing that Aptiva teaches storing the product key in a memory means.

The Applicant notes that there is no teaching in Aptiva of storing the product key in a

memory means.

The art fails to provide any motivation which would have suggested, to one of ordinary skill in the art, modifying Hoggarth or the combination of Hoggarth and Aptiva to include a feature of *writing the manually input product key information into the auxiliary memory*.

See *Karsten Mfg. Corp. v. Cleveland Gulf Co.*, 242 F.3d 1376, 1385, 58 USPQ2d 1286, 1293(Fed. Cir. 2001) ("In holding an invention obvious in view of a combination of references, there must be some suggestion, motivation, or teaching in the prior art that would have led a person of ordinary skill in the art to select the references and combine them in the way that would produce the claimed invention."); and *C.R. Bard, Inc. v. M3 Sys., Inc.*, 157 F.3d 1340, 1352, 48 USPQ2d 1225, 1232(Fed. Cir. 1998) (a showing of a suggestion, teaching, or motivation to combine the prior art references is an "essential evidentiary component of an obviousness holding").

In paragraph 13 starting on page 22 of the final rejection the Examiner holds "a system that inputs product key information must inherently store this information. It is well known in the art that the Windows™ operating system stores product key information in a registry file. The Information is inherently stored somewhere."

The Applicant disagrees. If the system product key for the Windows™ operating system were stored in the registry file, then anyone could go in to the registry file and look the information up. Microsoft is notoriously well known for trying to keep certain information secret, such as the product key for each operating system, in order to prevent piracy.

If it is well known, then the Examiner should provide a reference that teaches storing the Windows operating system's product key information on a personal computer. See MPEP §2144.03

The Examiner's suggestion of inherency is in error. *In Re Oelrich*, 666 f.2d 578# 581-82, 212 USPQ 323,326 (CCPA 1981) "The mere fact that a certain thing *may* result from a given set of circumstances is not sufficient [to establish inherency.]" *In re Rijckaert*, 28 USPQ 1953, 1957 (CAFC 1993) "a retrospective view of inherency is not a substitute for some teaching or suggestion supporting an obviousness rejection", citing *In re Newell*, 891 F.2d 899,901, 13 USPQ2d 1248, 1250 (Fed. Cir. 1989).

Accordingly, the rejection of claim 6 is deemed to be in error and should be withdrawn.

**I. Claim 7 was rejected under 35 U.S.C. §103(a), as rendered obvious and unpatentable, over Hoggarth in view of Aptiva as applied to claim 6, and in further view of Miura. The Applicant respectfully traverses this rejection for the following reason(s).**

Claim 7 calls for *deleting the product key information writing program after the product key information is written into the auxiliary memory.*

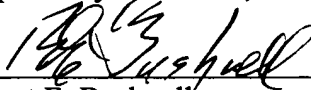
The traversal of the rejection is based on the same traversals as argued with respect to claim 14. Accordingly, the rejection of claim 7 is deemed to be in error and should be withdrawn.

The examiner is respectfully requested to reconsider the application, withdraw the objections and/or rejections and pass the application to issue in view of the above amendments and/or remarks.

Should a Petition for extension of time be required with the filing of this Response, the

Commissioner is kindly requested to treat this paragraph as such a request and is authorized to charge Deposit Account No. 02-4943 of Applicant's undersigned attorney in the amount of the incurred fee if, **and only if**, a petition for extension of time be required **and** a check of the requisite amount is not enclosed.

Respectfully submitted,

  
\_\_\_\_\_  
Robert E. Bushnell  
Attorney for Applicant  
Reg. No.: 27,774

1522 K Street, N.W.  
Washington, D.C. 20005  
(202) 408-9040

Folio: P56218  
Date: 8/10/05  
I.D.: REB/MDP